



Nastel XRay Express for Kafka

Docker Version

Version 1.0.0

Document Title: Nastel XRay Express for Kafka

Document Release Date: May 2020

Document Number: XR/KF 100.002

Published by:

Research & Development
Nastel Technologies, Inc.
88 Sunnyside Blvd, Suite 101
Plainview, NY 11803

Copyright © 2020. All rights reserved. No part of the contents of this document may be produced or transmitted in any form, or by any means without the written permission of Nastel Technologies.

Confidentiality Statement:

The information within this media is proprietary in nature and is the sole property of Nastel Technologies, Inc. All products and information developed by Nastel are intended for limited distribution to authorized Nastel employees, licensed clients, and authorized users. This information (including software, electronic and printed media) is not to be copied or distributed in any form without the expressed written permission from Nastel Technologies, Inc.

Acknowledgements:

The following terms are trademarks of Nastel Technologies Corporation in the United States or other countries or both: TransactionWorks, M6 AutoPilot, AutoPilot/IT, AutoPilot/Enterprise, M6 for WMQ, AutoPilot/WMQ, M6 Web Server, M6 Web Console, AutoPilot/WEB, MQControl, MQControl Express, AutoPilot/Transaction Analyzer, AutoPilot/WAS, AutoPilot/Transaction Monitor, AutoPilot/OS Monitor, Nastel Navigator, Nastel XRay.

The following terms are trademarks of the IBM Corporation in the United States or other countries or both: IBM, MQ, MQSeries, WebSphere, WebSphere MQ WIN-OS/2, AS/400, OS/2, DB2, AND AIX, z/OS

The following terms are trademarks of Hewlett-Packard in the United States or other countries or both: OpenView, HP-UX

Compaq, the Compaq logo, AlphaServer, Compaq Insight Manager, CDA, DEC, DECnet, TruCluster, ULTRIX, and VAX Registered in U.S. Patent and Trademark Office. Alpha and Tru64 are trademarks of Compaq Information Technologies Group, L.P in the United States and other countries

SNMPc, SNMPc, WorkGroup, and SNMPc Enterprise are trademarks of Castle Rock Computing in the United States or other countries, or both.

Sun, Sun Microsystems, the Sun logo, iForce, Java, Netra, N1, Solaris, Sun Fire, Sun Ray, SunSpectrum, Sun StorEdge, SunTone, The Network is the Computer, all trademarks and logos that contain Sun, Solaris, or Java, and certain other trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

InstallAnywhere is a registered trademark of ZeroG Software in the United States or other countries, or both.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>). The Jakarta Project" and "Tomcat" and the associated logos are registered trademarks of the Apache Software Foundation

Intel, Pentium and Intel486 are trademarks or registered trademarks of Intel Corporation in the United States, or other countries, or both

Microsoft, Windows, Windows NT, Windows XP, and the Windows logos are registered trademarks of the Microsoft Corporation.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

"Linux" and the Linux Logos are registered trademarks of Linus Torvalds, the original author of the Linux kernel. All other titles, applications, products, and so forth are copyrighted and/or trademarked by their respective authors.

SCO CUSA, SCO Doctor, SCO Doctor for Networks, SCO Doctor Lite, SCO Global Access, SCO MPX, SCO MultiView, SCO Nihongo OpenServer, SCO OK, the SCO OK logo, SCO OpenServer, SCO Open Server, SCO Portfolio, SCO POS System, SCO ToolWare, and The World Never Stops are trademarks or registered trademarks of Caldera International, Inc. in the U.S.A. and other countries, All Rights Reserved.

Oracle® is a registered trademark of Oracle Corporation and/or its affiliates

Other company, product, and service names may be trademarks or service marks of others.

Contents

| | |
|--|-----------|
| CHAPTER 1: INTRODUCTION | 5 |
| 1.1 HOW THIS GUIDE IS ORGANIZED | 5 |
| 1.2 HISTORY OF THIS DOCUMENT | 5 |
| 1.3 RELEASE NOTES | 5 |
| 1.4 INTENDED AUDIENCE..... | 5 |
| 1.5 TECHNICAL SUPPORT..... | 5 |
| CHAPTER 2: ABOUT NASTEL XRAY EXPRESS FOR KAFKA..... | 7 |
| 2.1 INTRODUCTION | 7 |
| 2.2 CONFIGURATION OVERVIEW..... | 7 |
| CHAPTER 3: FIRST STEPS | 9 |
| 3.1 INSTALLING NASTEL XRAY EXPRESS FOR KAFKA | 9 |
| 3.2 CREATING AN ON-PREMISE REPOSITORY | 9 |
| 3.3 YOUR FIRST LOG IN..... | 11 |
| 3.3.1 <i>Kafka Topics</i> | 12 |
| 3.3.2 <i>Kafka Messages Rates</i> | 13 |
| 3.3.3 <i>Kafka Partitions</i> | 15 |
| 3.4 CONFIGURING YOUR OWN DASHBOARDS..... | 16 |
| CHAPTER 4: USING THE NASTEL XRAY EXPRESS DATA COLLECTOR..... | 17 |
| 4.1 PRE-REQUISITES | 17 |
| 4.2 CONFIGURING NASTEL XRAY EXPRESS FOR KAFKA..... | 17 |
| 4.2.1 <i>Creating my_tnt4j.properties</i> | 17 |
| 4.2.2 <i>(Recommended) Using run_topic_list to listen to Multiple Topics</i> | 17 |
| 4.2.2 <i>(Optional) Using run to listen to specific brokers and topics</i> | 18 |
| 4.2.3 <i>Notes for Nastel XRay Express for Kafka Collector</i> | 18 |
| CHAPTER 5: IDENTIFYING TOPICS TO LISTEN TO..... | 21 |
| 5.1 TOPICS | 21 |
| INDEX | 23 |

This page intentionally left blank.

Chapter 1: Introduction

Welcome to the *Nastel XRay Express for Kafka*. This guide describes the installation, configuration and usability of the Nastel XRay Express for Kafka utility. Please review this guide carefully before using the product.

1.1 How This Guide is Organized

- [Chapter 1:](#) Identifies the users and history of the document and supplies support and reference information.
- [Chapter 2:](#) Contains a brief description of Nastel XRay Express for Kafka.
- [Chapter 3:](#) Provides instructions for downloading and deploying Nastel XRay Express for Kafka.
- [Chapter 4:](#) Explains how to run Nastel XRay Express for Kafka.
- [Chapter 5:](#) Discusses how to determine which topics to listen to.
- [Index:](#) Contains an alphanumeric cross-reference of all topics and subjects of importance.

1.2 History of This Document

| Table 1-1. Document History | | | |
|-----------------------------|-----------------|---------|-----------------------|
| Release Date | Document Number | Version | Summary |
| April 2020 | XR/KF 100.001 | 1.0 | Initial release. |
| May 2020 | XR/KF 100.002 | 1.0 | Update section 4.2.1. |

1.3 Release Notes

See `README.htm` files in your installation.

1.4 Intended Audience

This document is intended for personnel administering or using Kafka. The user should be familiar with:

- The target operating system environment.
- Procedures for installing software on the target platform.
- Using Kafka, such as Apache Kafka®, or other distribution such Confluent or IBM® Event Streams.

1.5 Technical Support

For technical support, visit the [Nastel Resource Center](#). You can contact us at <https://www.nastel.com/company/contact-us/>, or via live chat at the original registration link.

This page intentionally left blank.

Chapter 2: About Nastel XRay Express for Kafka

2.1 Introduction

The purpose of Nastel XRay Express for Kafka is to analyze applications using Kafka and determine their behavior. This document covers the specific use case of:

- 1) **Listening on Kafka topics**
 - Interrogate application behavior.

There are other samples in the express kit that implement additional use cases with Kafka and related components like ZooKeeper. These are not covered in this document. If these are of interest to you, see the website for other examples or contact us for information.

- 2) **Intercepting Kafka application calls to analyze application behavior**
 - Capturing calls and responses to Kafka requests.
- 3) **Gathering data from Kafka logs**
 - Identifying various conditions including errors and failures.
- 4) **Collecting metrics about Kafka performance**
 - Using JMX, mine the key Kafka performance indicators.
- 5) **Monitoring Kafka components**
 - Determine if they are running optimally.

2.2 Configuration Overview

There are four components that make up Nastel XRay Express for Kafka:

- 1) A repository for holding the Kafka data.
- 2) The graphical interface to present the data. This document provides a walkthrough of the basic capabilities in [3.3 Your First Log In](#) and additional details can be found in the Nastel XRay documentation.



At this point, sample repositories are available for viewing. Use these samples to preview the data you will be able to collect after completing steps 3 and 4 below.

- 3) An instance of the Nastel XRay collector running on at least one server. The setup and configuration of this is covered in [Chapter 4: Using the Nastel XRay Data Collector](#).
- 4) A Kafka broker running applications. You will need to configure which data to collect. For more information, see [Chapter 5: Identifying Topics to Listen to](#).

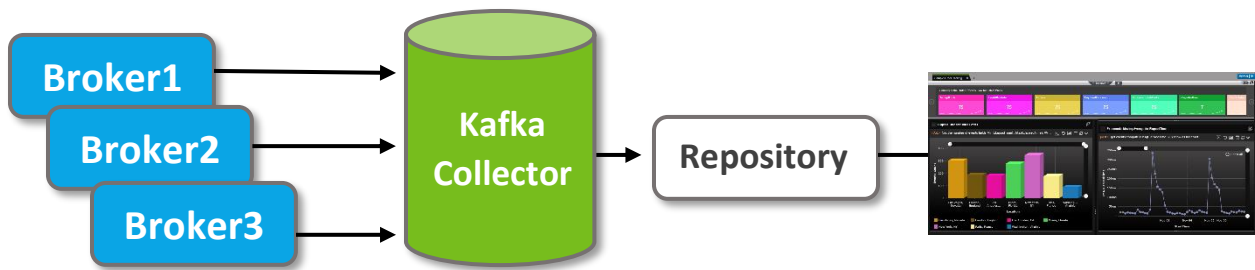


Figure 1. Configuration Overview

This document starts with setting up the repository since it is fundamental to all workflows. An introduction is then provided based on the sample data. The discussion is completed with how to add your data by configuring the collector and Kafka.



Figure 1a. Configuration Summary

Chapter 3: First Steps

3.1 Installing Nastel XRay Express for Kafka

The Nastel XRay Express zip file, **XRayExpress.zip**, contains everything you need. This file is available for download as part of the registration process. This file can be downloaded on Windows or Linux; use of Docker containers is not currently available on Windows. Data collectors can be run on either.

After unzipping the file, you will have the directory, *install_dir/XRayExpress*, containing samples and execution scripts used throughout this document.

3.2 Creating an On-premise Repository

This document covers installing a basic on-premise installation using Docker. This single server installation is targeted at introductory usage. All components can be scaled up to support massive environments, which is not covered in this document.

In order to run as a Docker container, the following **minimum** requirements must be met:

- Docker-supported Linux operating system
- 16 GB of RAM
- 4 virtual processors
- Docker installed and executable by the user without using `sudo` or other user switching options. Verify using `Docker -v` to report the Docker version. An example of the commands to install and setup Docker would be:
 - `sudo yum install docker`
 - `sudo service docker start`
 - `sudo usermod -a -G docker user`
(may require logoff and logon to take effect)

The *XRayExpress/docker* folder contains the scripts to run the Docker containers. If running on a different server than you downloaded to, copy the entire Docker folder, making the shell scripts executable (`chmod +x *.sh`).

There are 6 scripts including a menu to assist with setup. The scripts can be executed directly or from the menu.

To launch the menu, run **xray_menu.sh**. The following options are presented:

1. Deploy XRay Express containers
2. Display container status
3. Stop XRay Express containers
4. Start XRay Express containers
5. Clean XRay Express containers
6. Remove XRay Express images
7. Exit

Deploy Nastel XRay Express Containers

Executes the **new_xray.sh** script which downloads the Docker images, creates the containers and executes them. This is typically run once and then subsequent usage is done with stop and start options. This step will take approximately 5 minutes to complete.

If you have been provided alternative licenses, copy them to the **License** folder before running the **new_xray.sh** script.

You will be prompted for the XRay advertised host name. Enter the host (host name or IP address) that you are using for your XRay environment. You will also be prompted to select a dashboard for the sample repository. Select the Kafka dashboard to get the examples described in this document.

The script will suspend periodically to allow the configuration of the environment to complete, please let it finish completely.

Once complete, you can continue to [3.3 Your First Log In](#) using the hostname you provided above, such as: `http://myserver:8080/xray`

If you want to avoid prompting, you can run this script directly. In this you can also specify the IP/host as the first parameter (`./new_xray.sh 12.34.456.78`). You can specify the full name of a dashboard as a 2nd parameter. This must be in the dashboards folder and can be one of the samples provided or one obtained separately.

Display Container Status

This shows the status of the various containers using a Docker command. It includes all active containers on this system. Note that it is normal to see the init container stopped after initial deployment is completed.

Stop Nastel XRay Express Containers

Executes the **stop_xray.sh** script which will stop the containers. After stopping, they can be restarted with the start option.

Start Nastel XRay Express Containers

Executes the **start_xray.sh** script which will start the containers previously created. It can be used to restart a single container that has stopped since active instances are skipped.

Clean Nastel XRay Express Containers

Executes the **clean_xray.sh** script which deletes the containers from Docker and removes the working folder. This is typically done when you would like to start over. The containers must be stopped before running clean.

Remove XRay Images

Executes the **remove_xray.sh** script which removes the Docker images that were downloaded. The images must be cleaned before running the remove option.

Exit

Exits the script.

3.3 Your First Log In

In your browser, go to the URL provided and enter your user information to log in.

Login ID: Admin

Password: admin

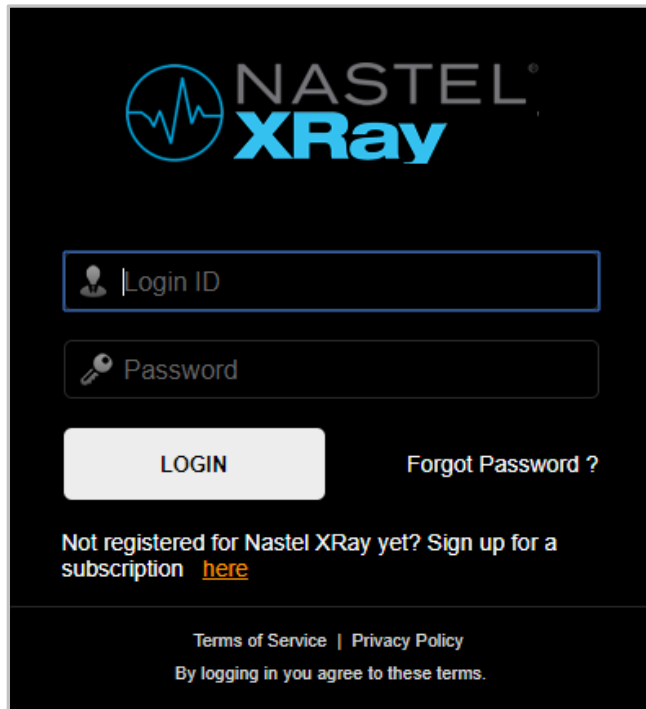


Figure 2. Logging In

When logging in for the first time, you will be given several options.

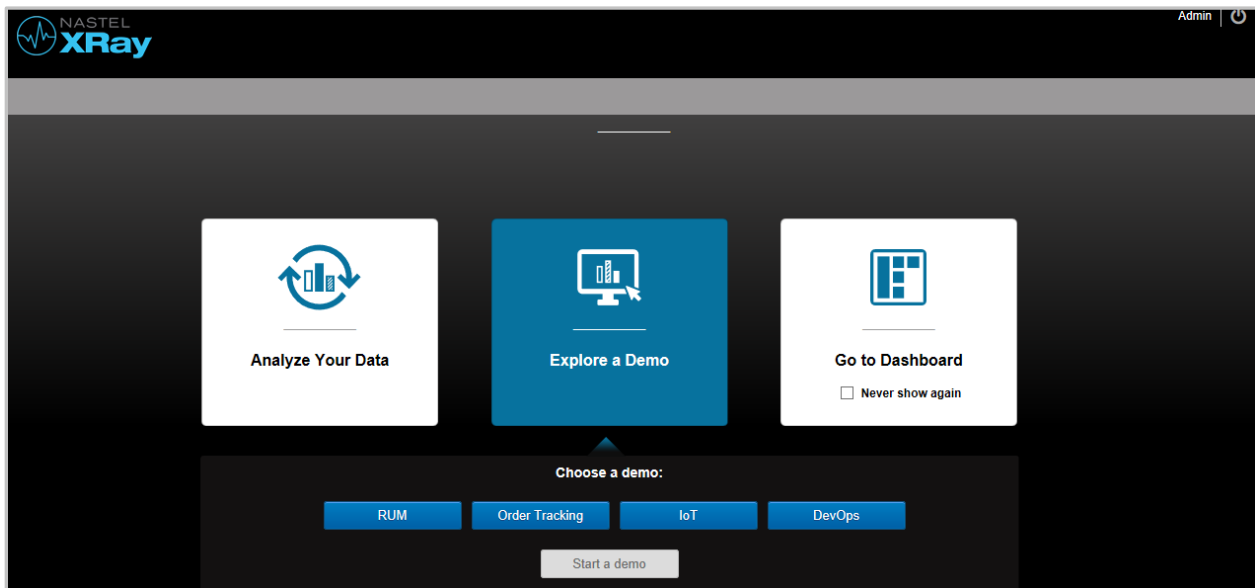


Figure 3. Landing Page

Select **Go to Dashboard** and optionally check the **Never show again** box to skip this landing page and go directly to your dashboard when logging in. You can review the demos now or at a later time by clicking **Explore a Demo**. The **Analyze Your Data** option is not applicable for the Kafka tracing data being processed.

To continue viewing the sample repository, select **Sample-Kafka_Tracing** from the **Repository** drop-down menu (see figure below).

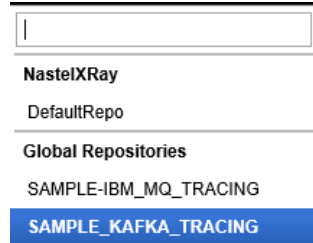


Figure 4. Select Sample Repository

To view the information, click on each tab as shown below

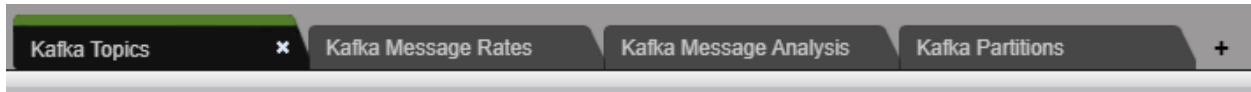


Figure 5. Dashboard Tabs

3.3.1 Kafka Topics

This section will walk through the various sections of the page (referred to as viewlets).

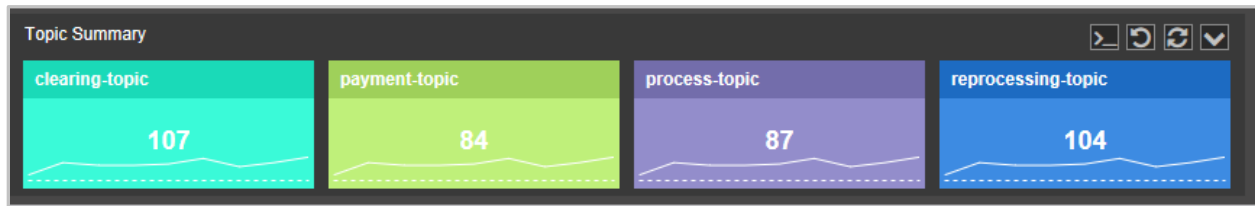


Figure 6. Topic Summary

At the top is a summary viewlet. In this case, this shows a summary of all of the Kafka topics captured. In this sample, there are messages produced to the clearing-topic and payment topic, process-topic and reprocessing-topic.

If you want to see all of the publications related to a specific topic, you can just click on the summary value.

| | StartTime | KafkaTopic | Partition | Offset | Message | Value Size | Key Size | KafkaLabel |
|--------------------------|----------------------|----------------|-----------|--------|------------------------|------------|----------|------------|
| <input type="checkbox"/> | 7/29/2019 2:11:34 PM | clearing-topic | 0 | 211 | <?xml version="1.0"... | 454 | -1 | MyBroker |
| <input type="checkbox"/> | 7/29/2019 2:11:34 PM | clearing-topic | 1 | 210 | <?xml version="1.0"... | 467 | -1 | MyBroker |
| <input type="checkbox"/> | 7/29/2019 2:11:38 PM | clearing-topic | 0 | 212 | <?xml version="1.0"... | 510 | -1 | MyBroker |
| <input type="checkbox"/> | 7/29/2019 2:11:51 PM | clearing-topic | 2 | 184 | <?xml version="1.0"... | 486 | -1 | MyBroker |
| <input type="checkbox"/> | 7/29/2019 2:12:01 PM | clearing-topic | 1 | 211 | <?xml version="1.0"... | 527 | -1 | MyBroker |

Figure 7. Kafka Message Detail

To close the console, click on the **x** after **Event Details** or on the Console identifier.

Below the Summary is a pie chart showing the distribution of the topics used.

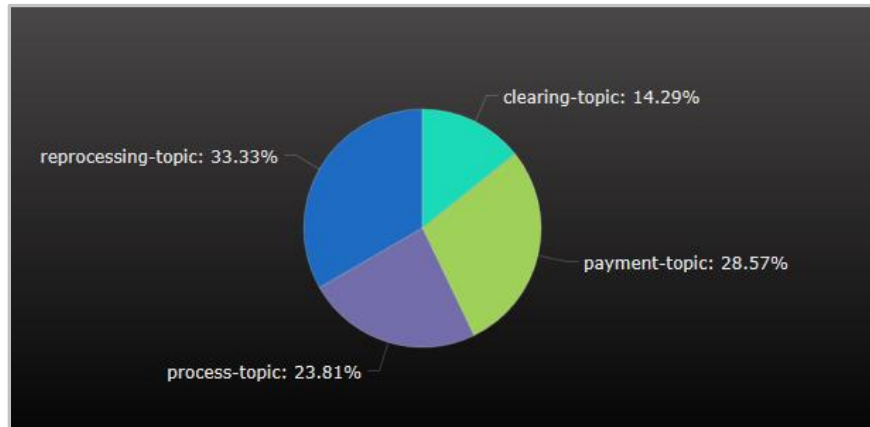


Figure 8. Kafka Topic Breakdown

3.3.2 Kafka Messages Rates

The first viewlet on this dashboard shows a chart of the topics produced per minute. For larger periods of time, you can also group bucketed by hour or days.



Figure 9. Kafka Message Rates

You can click on the topics in the legend to toggle them showing on the graph.

At the bottom of the page, the viewlet shows a breakdown by message size, broken down into groups of 100.

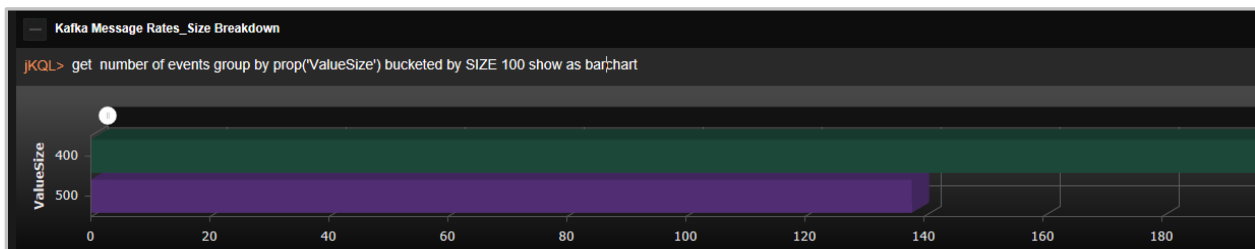


Figure 10. Kafka Message Sizes

If you want to view the specific messages of this size, you can just click on the respective bar. For example, when clicking on the 400 range bar, you get the list of topics in this size range.

| | StartTime | KafkaTopic | Partition | Offset | Message | ValueSize | KeySize | KafkaLabel |
|--------------------------|----------------------|----------------|---------------|--------|-------------------------|-----------|---------|------------|
| <input type="checkbox"/> | 7/29/2019 2:11:32 PM | process-topic | 1 | 242 | <?xml version="1.0" ... | 462 | -1 | MyBroker |
| <input type="checkbox"/> | 7/29/2019 2:11:32 PM | process-topic | 0 | 254 | <?xml version="1.0" ... | 454 | -1 | MyBroker |
| <input type="checkbox"/> | 7/29/2019 2:11:34 PM | clearing-topic | | 211 | <?xml version="1.0" ... | 454 | -1 | MyBroker |
| <input type="checkbox"/> | 7/29/2019 2:11:34 PM | clearing-topic | payment-topic | 210 | <?xml version="1.0" ... | 467 | -1 | MyBroker |

Figure 11. Viewing Specific Messages

To see the data in a different way, right click on the dashboard title and select configure:

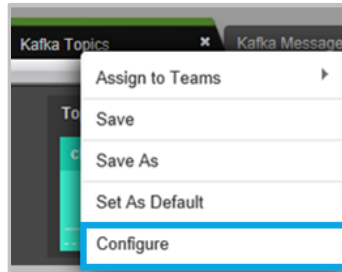


Figure 12. Select Configure

Under **Schema**, select **KafkaIDOC**.

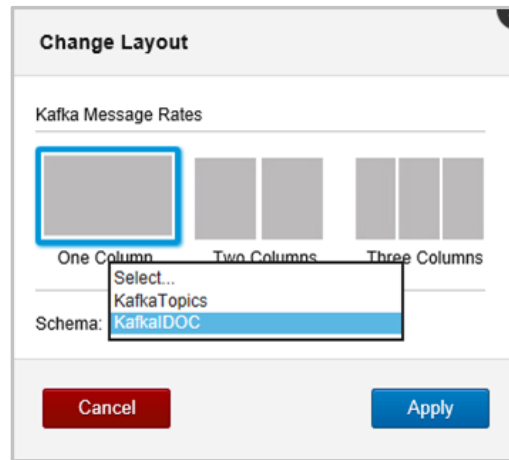



Figure 13. Select KafkaIDOC

Because the sample was made from a known set of data, we also configured the collector to extract specific fields out of the message content as separate properties. This Schema is designed to show those fields.

Now refresh the display by clicking on the refresh icon  or you can close the console and click on the message size of 400 again. Now the console window shows a different set of columns as shown below, tailored to the data captured.

JKQL> Get Event where Properties('ValueSize') between '400' and '499'

| | StartTime | IDOC | Currency | Value | Status | KafkaTopic |
|--------------------------|----------------------|---------|----------|-------|-------------------|--------------------|
| <input type="checkbox"/> | 7/29/2019 2:17:52 PM | ID1 | EUR | 37002 | QUOTE_REJECTED | reprocessing-topic |
| <input type="checkbox"/> | 7/29/2019 2:20:03 PM | ID10027 | EUR | 23204 | QUOTE_REJECTED | reprocessing-topic |
| <input type="checkbox"/> | 7/29/2019 2:16:19 PM | ID10058 | USD | 21193 | PAYMENT_REQUESTED | payment-topic |
| <input type="checkbox"/> | 7/29/2019 2:14:33 PM | ID10104 | USD | 23987 | PAYMENT_REQUESTED | payment-topic |
| <input type="checkbox"/> | 7/29/2019 2:14:44 PM | ID10146 | EUR | 17756 | PAYMENT_REQUESTED | payment-topic |
| <input type="checkbox"/> | 7/29/2019 2:19:24 PM | ID10167 | USD | 6834 | PAYMENT_CLEARED | clearing-topic |

Figure 14. Event Details

These fields could have been displayed as graphical as well, showing the total value or breakdown by currency. As a demonstration, click on the + next to **Console**. This opens up a temporary viewlet. Enter the following in the JKQL> field:

Get Event field sum(Value) group by Currency show as colchart

The following chart showing the total value by the four currencies appears:



Figure 14A. Currency Totals

To close, click on the X or on the console label.

3.3.3 Kafka Partitions

The first viewlet on this dashboard shows a breakdown of the topics collected and the breakdown of the partitions where those messages were stored. You should see a uniform breakdown of the events across the topics. When using your own keys, the distribution may be less uniform depending on your key structure.

The screenshot shows a dashboard titled "Kafka Partitions_Partition Analysis". Below the title is a command prompt: `jKQL> Get number of events group by map('KafkaTopic'), Partition show as scorecard`. The main content is a table with three columns: "KafkaTopic", "Partition", and "Events Count".

| KafkaTopic | Partition | Events Count |
|--------------------|-----------|--------------|
| clearing-topic | 0 | 23 |
| | 1 | 34 |
| | 2 | 26 |
| | 3 | 30 |
| payment-topic | 0 | 26 |
| | 1 | 34 |
| | 2 | 28 |
| | 3 | 28 |
| process-topic | 0 | 25 |
| | 1 | 35 |
| | 2 | 32 |
| reprocessing-topic | 0 | 53 |
| | 1 | 54 |

Figure 15. Kafka Partition Analysis

The second viewlet breaks down the keys being used. A value of -1 indicates no key which means that Kafka will just distribute across the partitions.

The screenshot shows a dashboard titled "Kafka Partitions_Keysize Breakdown". Below the title is a command prompt: `jKQL> get number of Event group by prop('KeySize') bucketed show as table`. The main content is a table with two columns: "KeySize" and "Events Count".

| KeySize | Events Count |
|---------|--------------|
| -1 - -1 | 400 |

Figure 16. Kafka Keysize Analysis

Note that there are other sample repositories available to view from the **Repository** drop-down menu.

3.4 Configuring Your Own Dashboards

Now that you have seen the previous examples, you will want to configure your repository for your Kafka data. For this section, return to your own repository by selecting it from the repository dropdown. This will be **DefaultRepo**.

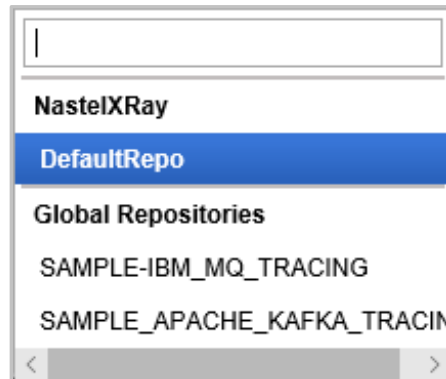


Figure 17. Select Your Repository

Chapter 4: Using the Nastel XRay Express Data Collector

4.1 Pre-requisites

If you have not already unzipped the **XRayExpress.zip** file, download and unzip it. Copy it to the system you want to run the collector on. See [Section 3.1](#) for more information.

Nastel XRay Express for Kafka requires:

- Target Kafka services running
 - Nastel XRay Express for Kafka can be installed on the Kafka server or as a remote client.
- **Set JAVA_HOME to the installed version of Java.**

4.2 Configuring Nastel XRay Express for Kafka

4.2.1 Creating my_tnt4j.properties

In the run folder, the member **XRay_tnt4j.properties** is a template that can be used to create custom properties for your usage. The easiest way to do this is to run **XRay_setup.sh** or **XRay_setup.cmd**. You can either supply the parameters or it will prompt you for them.

Syntax: XRay_setup.sh URL TOKEN

URL: your local URL for on-premise

Example: http://myserver:6580

TOKEN: DefaultToken provided during registration

Example: 9ad9a6aa-bb5d-4ef4-a7fa-Et0835b6b8dc DefaultToken

On completion, you will have a new file called **my_tnt4j.properties** in the run folder.

Alternatively, you can manually copy **XRay_tnt4j.properties** to **my_tnt4j.properties**, making the changes as outlined.

4.2.2 (Recommended) Using run_topic_list to listen to Multiple Topics

If you are running on Linux and want to listen to a number of topics for the same broker on Linux, use this option. In the folder **run/kafka-client**, edit the member **topics.txt**. This is a list of topics that you want to capture data from. Change the sample names included to the list of topic names required and save the file.

Execute the **run_topic_list.sh** to start capturing messages produced to these topics. The format of the command is:

```
run_topic_list.sh connection_info topic_list_file label
```

where,

connection_info is the broker connection string (default: localhost:9092)

topic_list_file is the file described above (default: topics.txt)

label is any identifier you want included with the data collected (default: same as connection_info)

Examples:

- run_topic_list.sh
- run_topic_list.sh kafka_server:9092
- run_topic_list.sh kafka_server:9093 mytopics.txt broker123

Hint: If you want to listen to all Kafka topics? Pipe the output from a `topics --list` command as the `topic_list` file. You can use as is or edit it to remove topics you do not want to trace.

4.2.2 (Optional) Using run to listen to specific brokers and topics

To listen to various topics in different brokers, use this option. In the folder `run/kafka-client`, edit the member `XRayKafka_parser.xml`:

1. Specify the topic string that you want to listen to.

```
<property name="Topic" value="YOUR_KAFKA_TOPIC"/>
```
2. Change the following as defined by your Kafka setup.

```
<property name="bootstrap.servers" value="YOUR_KAFKA_HOST:YOUR_KAFKA_PORT"/>
```
3. For this use case, you want the `group.id` to be a unique value so that it does not conflict with any application consuming messages from this topic.

```
<property name="group.id" value="0"/>
```
4. All other changes are optional and can be completed at a later time.
5. If you want additional topics, repeat the `<stream>` section as many times as required, setting a unique value for stream name for each. A single instance of the Nastel data collector can collect data from many Kafka topics and brokers. Change the name of the client to be unique as well.

```
<property name="client.id" value="XRay-kafka-consumer"/>
```
6. Save your changes

To start the data streaming process, from the `run/kafka-client` folder, execute either `run.bat` or `run.sh`. Using the options supplied, it will connect to the broker(s) specified and wait for data.

Now return to your repository to view data based on the data from your topics ([Section 3.3.1](#)).

4.2.3 Notes for Nastel XRay Express for Kafka Collector

If you see any error messages during startup, correct them and repeat. Additional logging information can be found in the logs folder, in `XRay-streams.log`.

When you add additional Kafka topics or brokers to the `XRayKafka_parser.xml` or make any other changes, you will need to stop and restart the collector.

At this point, if your configuration is correct, you should start receiving data in your repository.

If you do not want to capture message data, edit the sample parser and remove the Message field. If you want to parse specific data elements as properties rather than a single message as shown in the examples above, see the samples for versions for XML, JSON, CSV and other formats.

This page intentionally left blank.

Chapter 5: Identifying Topics to Listen to

5.1 Topics

One of the advantages of Kafka is that you can listen to any topic with no impact to the applications. Reasons for listening to topics might include:

- Identifying the usage patterns of the application
- Determining if partition algorithms are effective

Determining which topics to listen to may require some planning and tools like AutoPilot for monitoring Kafka.

You can listen to all topics if required, by using the topics --list request and using that list or a subset as input into the Kafka collector discussed in chapter 4.

This page intentionally left blank.

Index

D

Dashboards.....12, 16
Document History.....5

I

Installing9

K

Kafka
 Messages Rates13
 Partitions15
 Topics12

M

my_tnt4j.properties.....17

O

On-premise Repository.....9
Overview.....7

P

Pre-requisites
 Nastel XRay Data Collector.....17, 21

R

README files5

T

Technical Support5
Topics.....21